Fall 11-1-2004

# Madison Christian School Administrative Database

Pamela Rowland
*Dakota State University*

# Madison Christian School Administrative Database

A Project Report submitted to the

Graduate Faculty

By

Pamela Rowland

In partial fulfillment of the requirements for the degree of

Master of Science in Information Systems

Dakota State University

November 2004

DAKOTA STATE dsu UNIVERSITY

# MSIS
# PROJECT APPROVAL FORM

Student Name: Pamela F. Rowland

Expected Graduation Date: December 2004

Master's Project Title: Madison Christian School Administrative Database

Date Project Plan Approved: May 2004

Date Project Coordinator Notified and Grade Submitted: 12/04/04

Approvals/Signatures:

Student: Pamela Rowland Date: 12-02-04

Faculty supervisor: Stephen Hedaby Date: 12/04/04

Committee member: [signature] Date: 12/04/04

Committee member: [signature] Date: 12/07/04

Revised 1/28/03

**Abstract**

The need for a database system at Madison Christian School (MCS) is examined. An investigation of current processes identifies the need for stored data that is easy to access and easy to use. Currently paper data is stored in a file cabinet. It is not easy to manipulate, and reports must be generated by hand. MCS needed a database that a teacher with little computer skills could easily access and use while considering that an administrator would use the database for data manipulation and storage. Microsoft Access is used to develop the database as it is an easy-to-use interface that MCS has installed currently on their system. Future growth is taken into account, and a similar database is created in Oracle. Oracle was linked to Access as an example of how MCS could use it as a server-side application in the future.

## Table of Contents

## List of Figures

# I. Introduction

Madison Christian School is a K-8 private school located in Madison, SD. MCS desires to have a relational database to be used by an MCS administrator and/or head teacher. Information in the database will be about the MCS people – faculty, staff/board members, and students. It will also contain information about subjects taught including what books are used for which subjects. The administrator will be able to use the information to determine what subjects are taught by teachers, to review books used for each subject, and to contact individuals.

Currently MCS uses a file cabinet and paper data. This data is not easy to manipulate or to generate reports. Currently reports are compiled and typed manually as the need arises.

The Access database developed provides an easy-to-use environment that the school will use on a daily basis. An easy-to-use interface that the users could learn quickly was critical. The school currently has the Access program, and it was important that there were no additional costs to the school for the database. The Oracle database was developed to connect to the Access database. This was done to provide for future growth at MCS. If the need should arise, Oracle is ready to be implemented. During the development process, it was necessary to be forward-looking. The school anticipates much growth in the coming years, and Oracle may be a system they will look at in the future.

Developing the database with Access and Oracle allowed the project coordinator the opportunity to do a comparison of the programs.

## II. Organizational Information

**Organization Objectives:**

The principle organizational objective of the Madison Christian School is to provide a quality, cost effective, education in a Christian environment for grade K-8 students.

**Organizational Structure:**

Madison Christian School (MCS) is a 501 c (3) non-profit organization, governed by an elected and appointed school board. Elections are held annually by the Madison Christian School Association. The school is currently in its fourth year of formal operations.

The school board oversees all administrative functions including education, finance, recruitment, facilities, and public relations in accordance with a federally-approved constitution and by-laws.

A school administrator will, in the near future, administer the daily education functions. This position has yet to be filled, so the board president fulfills this role temporarily with the assistance of the school board members.

Teaching faculty and other staff are hired and assigned by the school board.

Students are appraised and admitted using a formal application process.

**Description of Operations:**

- **Education** – All functions of the school are intended to support the educational function. The administrator is specifically responsible for the following concerns:
    1. Ensuring that all classes meet according to schedule;
    2. Guaranteeing that the school employs qualified teachers;

3. Assuring that the teachers adhere to a prescribed curriculum in an organized Christian environment;
4. Facilitating positive relations among faculty, staff, students, families, and churches.

Other aspects of the educational functions of the school include keeping detailed records of student achievement, providing access to quality materials and opportunities, and conscientiously growing a reputation of educational excellence in the local community.

- **Finance** – Adherence to strict accounting practices is essential to the reputation and success of this educational endeavor. As such, open and accurate accounting methods detailing all incoming and out-going finances are vital to our continued future growth. Typical receivables are tuition, fees, and donations. Payables are wide ranging and include such elements as salary, equipment cost, maintenance and repair, utility payments, as well as textbooks and other teaching supplies. At this time the school board treasurer deals with all financial recording. This individual has a detailed software package that is working effectively.
- **Recruitment** – A school cannot function without students. Recruitment efforts have thus far involved the coordinated efforts of numerous area congregations, local fund- and awareness-raising endeavors, and the concerted efforts of a great many volunteers. A public relations campaign has been launched to help the school to identify and recruit students. Other communal efforts have included research efforts to better understand financial, social, and political barriers to this

parochial undertaking. The school board has actively sought to eliminate the real or perceived barriers to this project.

- **Facilities** – The current school facilities are being leased from West Center Baptist Church of Madison. The space limitations of the current facilities necessitate planning for MCS to have its own building in the near future. Thus, facilities management includes the negotiation and management of current lease agreements and also involves working closely with the financial arm of the organization in all fund-raising efforts. Redesign, maintenance, and repair of the facilities all require dedicated personnel, whether they are hired professionals or part of a local volunteer force.
- **Public Relations** – Vital to the success of the Madison Christian School is the warm, continued support of the local Madison community. The cultivation of a healthy relationship with the community involves the coordinated efforts of churches, parents, students, and community leaders. The Madison Christian School must strive to create a reputation for the academic and spiritual quality of its students. A key factor in the school's public relations function will be information about churches, schools, the local business community, and its own students.

**Madison Christian School Organizational Members**

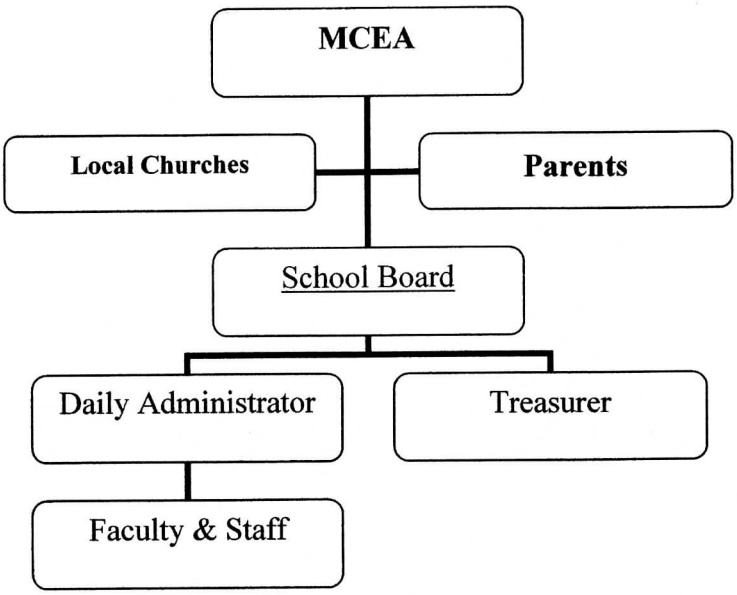



Figure 1: MCS Organization Members

**Business Rules:**

The business rules for MCS are numerous, as the laws governing private parochial institutions are necessarily quite exhaustive. A brief summary of the more noteworthy rules of the institution is as follows, though not all of those mentioned here are relevant to the database needs of the school:

- Madison Christian Education Association has membership rules.
- Local church participation follows specific guidelines.
- Election of the school board is governed by specific rules.
- Qualifications for faculty are set by school board procedures.
- Student applications follow specific guidelines.

- Tuition and fees conform to specified rules.
- Teaching curriculum follows accreditation rules of an independent Christian school association and follows graded standards of the State of South Dakota.
- The school calendar details a specific beginning and end to the school year, number of required class days, and planned holidays which follows standards set by the State of South Dakota.
- The class day has a set time schedule.
- Students must follow a written code of conduct.
- Donations are receipted according to IRS 501 c (3) regulations.
- Payroll reports and withholdings follow IRS tax law.
- A lease agreement for use of the building has specific rules.

## III. Problem Identification

**Interview:**

The first step in the development of this project was to interview the head teacher. MCS is in its fourth year of operation and is a relatively small school, so it does not have an administrator at this point. The head teacher and the school board president/administrator are the key individuals for using the database at the present time. A copy of the interview questions is in APPENDIX A of this paper. Key results from the interview are as follows:

- Quality education – the head teacher can not be bogged down with menial tasks of bookkeeping, filing, or ordering materials. Teaching staff needs to be able to concentrate on teaching.
- Easy-to-use – the system needs to be extremely user friendly.
- Information availability – Administration and teachers need to be able to get the information they need quickly and easily.
- Information needed:
    - Student list by grade
    - Teacher list by subjects they are teaching
    - Book information

## IV. System Request

The school board requested that a database be developed to aide the teachers and administration in the daily activities of MCS.

The specific requirements are as follows:

- The proposed system should provide student, staff, and faculty information.
- It should provide subjects taught in each grade.
- It should provide the books used and publisher's name and contact information.
- It should generate reports, such as what students are in a particular grade level and what teacher is teaching a particular subject.

## V. The Development Process

The development of the MCS database followed closely to the life cycle development approach. It followed the steps of planning, analysis, design, and implementation. The planning stage was critical to the development of an appropriate database to meet the needs of MCS. An initial interview was conducted with many follow-up interviews as the development progressed. These interviews allowed the development of a database that would be easy to use with the right information for the end user.

Developing the correct ER diagram and relational model allowed for a good transition into the development of the database. After many drafts, the final ER diagram proves to be effective. It is shown in APPENDIX D.

Microsoft Access was determined to be the best platform to use for the database. Oracle was also implemented to allow for future growth of the school's data.

Initial implementation of the database has been successfully. The users have been pleased with the data and how easily it can be accessed. To properly analyze the database it will take a full year of use to see if all needs are met. The project coordinator will continue to work with MCS in developing and adding to this database as needs arise.

## VI. Objectives and Deliverables

The goal of this project was to provide a database that stored MCS data that could be manipulated to produce reports and queries. MCS needed a very easy-to-use system that would provide information needed quickly and effectively.

The best program to use for MCS was Microsoft Access. The school currently has the program, and it could be implemented easily with no cost to the school. Oracle was also implemented to allow for future growth of the school's data.

The only changes made after the initial project planning document were changes to the ER diagram and relational model. After interviewing the end users, some changes were necessary to provide the data needed for the school.

## VII. Database Design

The database was designed after several ER diagrams and relational models were created. The first ER diagram that appeared in the planning document did not meet specific needs of MCS, and in a few areas it covered too much information. The first diagram showed an entity type of PEOPLE as a supertype after the normalization process. Using this entity was found to be too broad for security purposes. It was decided to break that down into FACULTY, STUDENT, and STAFF entity types. This way if an end user was not allowed access to student records, there could be security measures put into place.

In the normalization process, it was recognized that there is a transitive dependency in city, state, and zip within each of the entities of FACULTY, STUDENT, and STAFF. However, it was decided not to normalize these attributes. Currently all the students, faculty, and staff at MCS are from Madison, SD, and it is anticipated that a majority of students will be in the future as well. Rather than normalize these attributes it was decided to use a default value for each city, state, and zip.

Another change was in the area of SUBJECT. After interviewing the teachers and working with MCS in preparing for the school year, it was evident that publishers' names needed to be attached to each book for each grade. Often a book needed to be ordered, and that information needed to be readily available. Having the school's account numbers and the order number for the books are of great assistance to the school.

Teachers indicated that they did not need to store student's grades in this database, and in fact did not want to do so. Being the small school that MCS is currently,

room location for a subject was not needed. That can easily be added in the future if a larger building is acquired.

The ER diagram and Relational model are located in the APPENDIX D.

## VIII. Microsoft Access Implementation

Microsoft Access was determined to be the main application for the database. Microsoft Access is not designed to be used as a server database, and currently MCS does not utilize a server. MCS does have Access installed on their system allowing an easy, no-cost development to take place. Several designs were tried and tested with the end users. Parts of the final design are shown below.

A switchboard is used to help users get to the information they want quickly. The user simply has to click on the command button to go to the correct form.

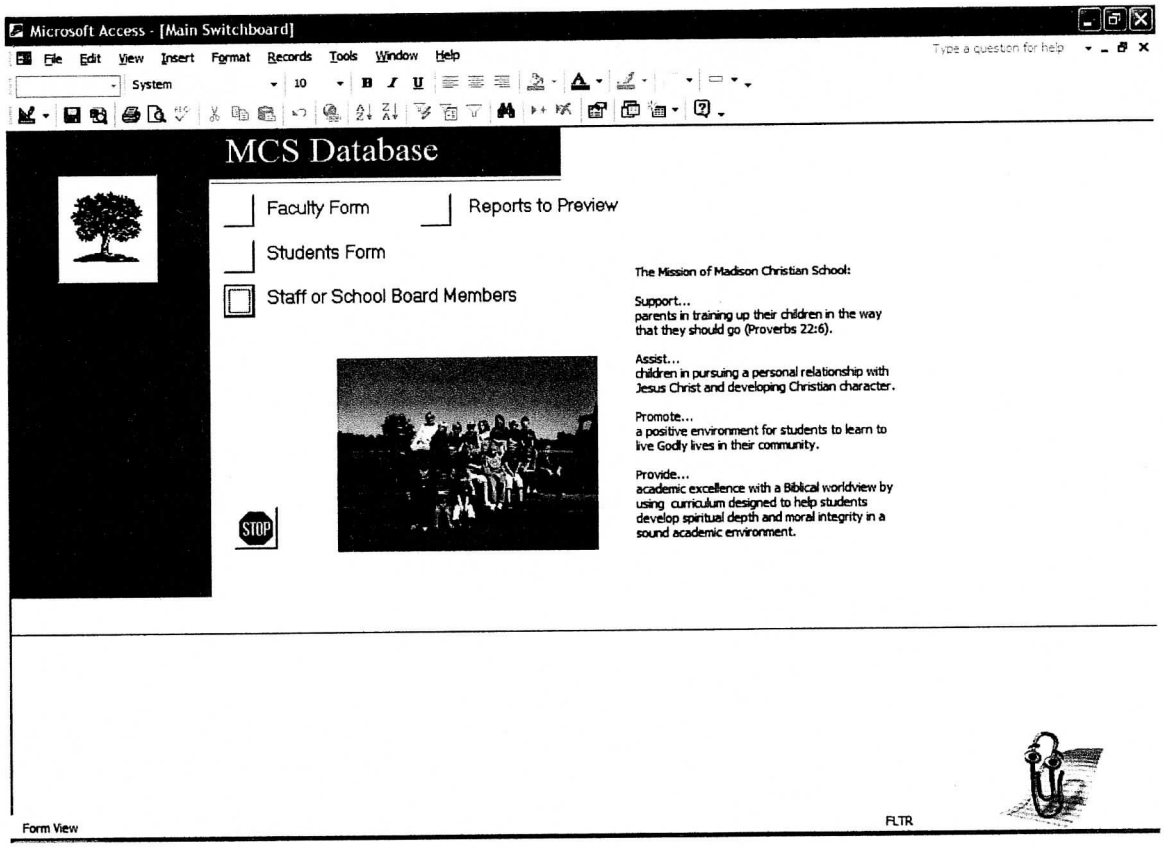



Figure 2: Switchboard

Administration can access Faculty, Student, or Staff/School Board Member records by clicking on the appropriate button. New members can be added in the form

view. The Student Form and Staff/Board Members Form are similar so only one is shown in Figure 3.

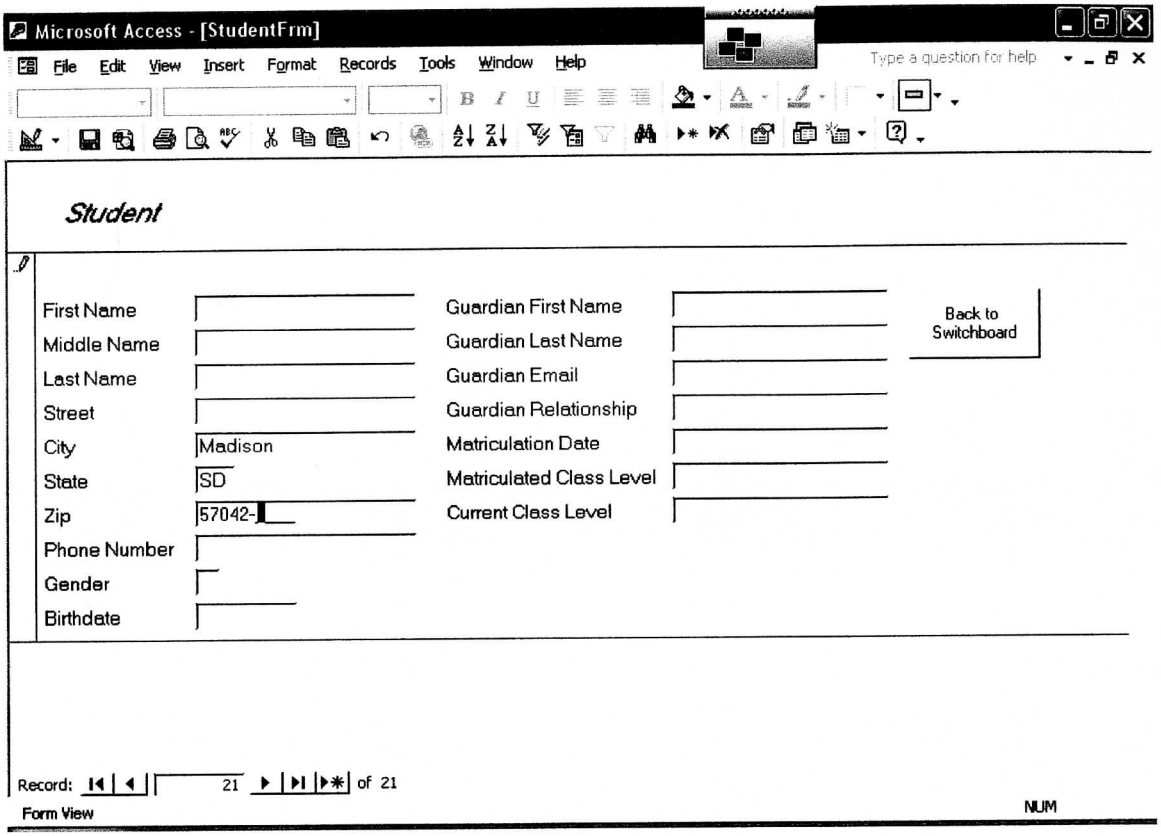



Figure 3: Student Form

The Faculty form is also tied to what classes the faculty member teaches via use of a subform connected to a Teaches Table. This is shown in Figure 4 below.

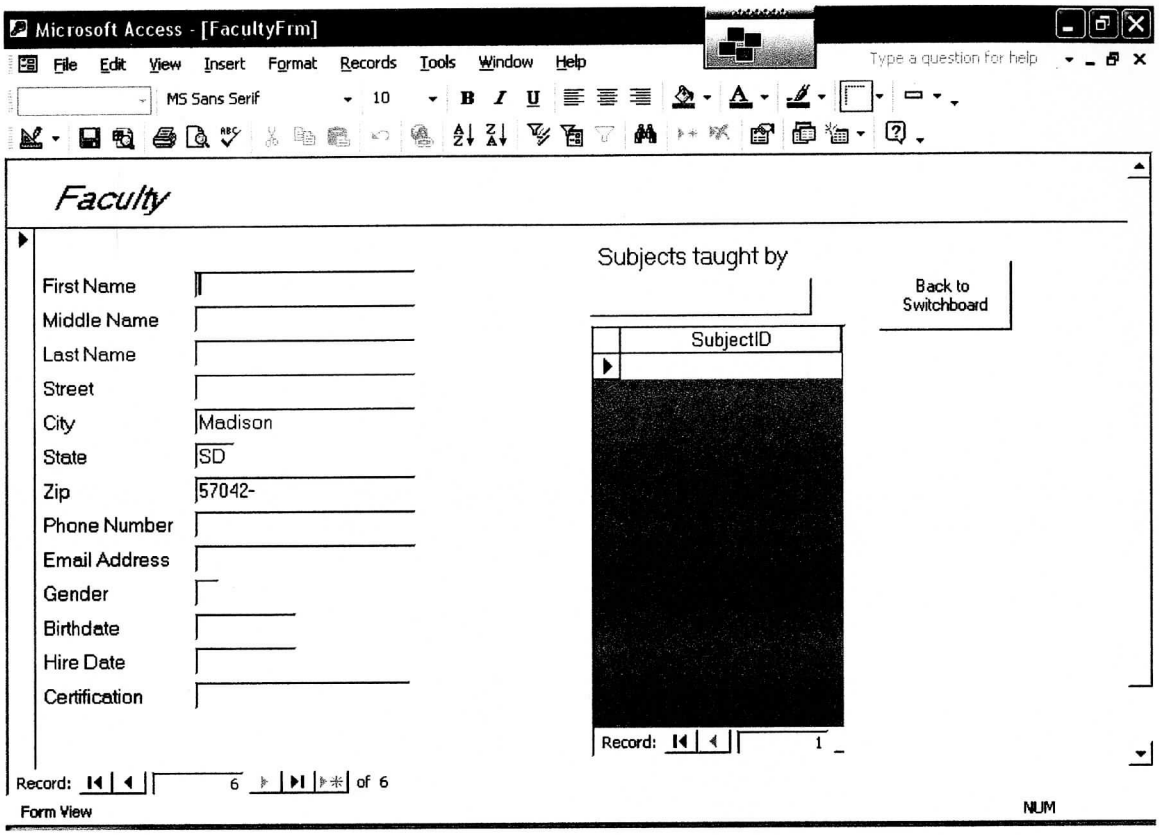



Figure 4: Faculty Form

The Reports form leads the user to a number of commonly used reports.

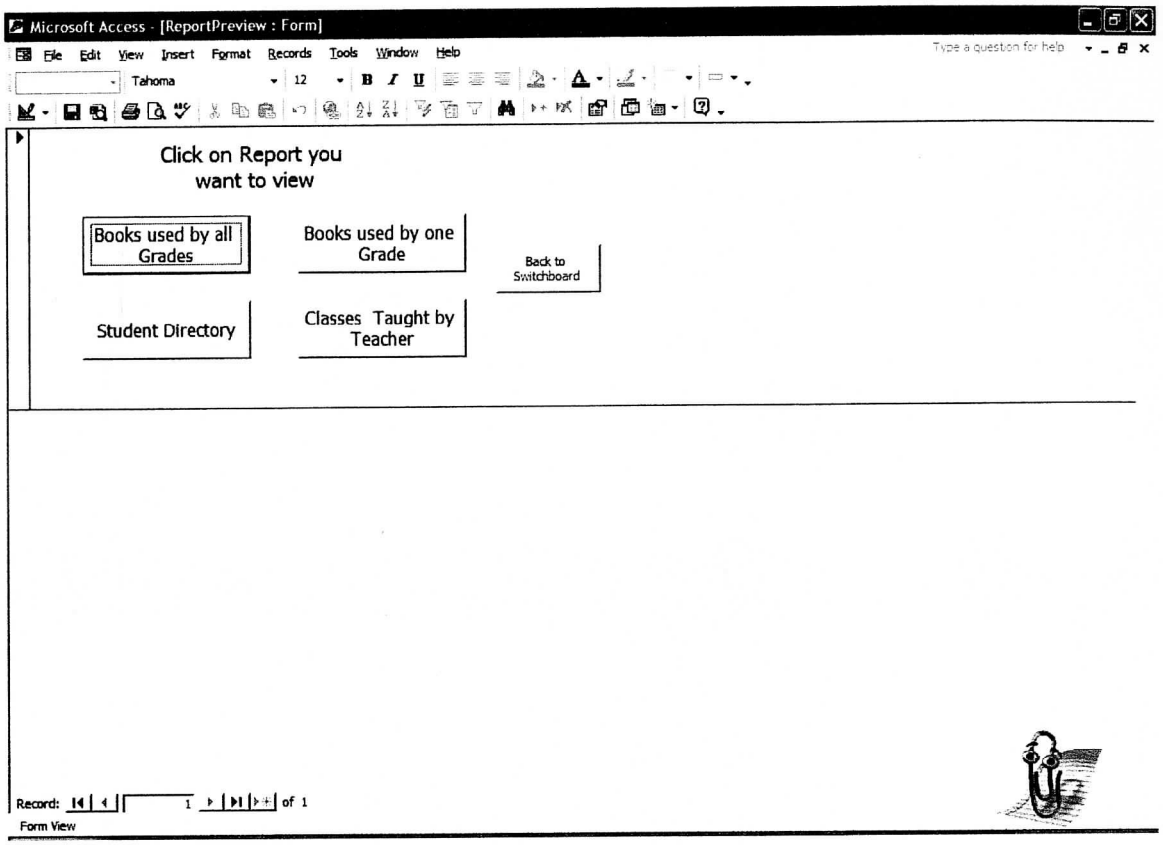



Figure 5: Reports' Form

A description of each report is as follows with an example of each report in APPENDIX B of this paper:

- Books used by all Grades
    - This report lists all the books used by grade level. It shows the name of the subject, the book, the publisher, and the book ID.
- Books used by one grade
    - This report is similar to the above report; however, it limits the report to one grade.
- Student Directory

- By grade this report lists the student's name, address, phone number, and birth date.

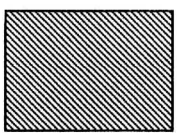
- Classes taught by Teacher
  - This report breaks down what each teacher teaches.

Most reports were created with the use of a query. As an administrator or teachers have the need and required clearance, they would be able to run their own queries and reports.

## IX. Oracle 9i Implementation

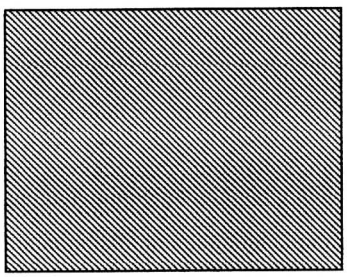
Oracle 9i was implemented with the same data as was used in Access. This process was done to help the school determine future needs. Oracle could be used in a server environment as the school grows with Access at the front-end. By utilizing Access on the front-end, the administration and faculty would still have an easy-to-use environment to work in.

Oracle 9i gave some problems initially with the installation of the software. The software was downloaded from the internet, which took several hours. After what was thought to be a successful download, an installation process was started and failed. Finally a student edition of Oracle 9i was made available and was installed. Code was written in Notepad and copied into SQL Plus to create the tables and populate them. Then a few sample queries and reports were created and run. Because this is not the software the end user will be using, the Form Builder was not installed. APPENDIX C shows the SQL code for the tables. Below is output from three sample queries and one report.

```
SQL> --Query 1 Select all students first and last names in the first grade
SQL>
SQL> SELECT student_id, first_name, last_name
  2  FROM student
  3  WHERE current_class_level='first';

STUDENT_ID FIRST_NAME          LAST_NAME
---------- -------------------- -------------------------

       101 Dharma               

       102 Brett

SQL> --Query 2 select subject name, book used, and publisher for grade 1 math
SQL> SELECT subject_name,book_title,publisher_name
```

```
  2  FROM subject,book,publisher
  3  WHERE subject.subject_id=book.subject_id
  4  AND book.publisher_id=publisher.publisher_id
  5  AND subject_name='1Math';

SUBJECT_NAME          BOOK_TITLE                    PUBLISHER_NAME
--------------------  ----------------------------  --------------------

1Math                 First Math                    ACSI


SQL> --Query 3 select student name, guardian's name, and phone number
SQL> SELECT first_name, last_name, guardian_first_name, guardian_last_name,
phone_no
  2  FROM student
  3  order by last_name;

FIRST_NAME          LAST_NAME             GUARDIAN_FIRST_NAME
------------------- --------------------- -------------------
GUARDIAN_LAST_NAME  PHONE_NO
------------------- ----------
Camora              [redacted]            Josh

Dharma                                    Josh

Brett                                     Dywane

FIRST_NAME          LAST_NAME             GUARDIAN_FIRST_NAME
------------------- --------------------- -------------------
GUARDIAN_LAST_NAME  PHONE_NO
------------------- ----------
Morgan              [redacted]            Dean

Amanda                                    Loren

Thomas                                    Loren
```

```
SQL> --Creating a report
SQL> TTITLE LEFT "Student Phone Directory" -
> Left 'Student'
SQL> SET LINESIZE 100
SQL> SET PAGESIZE 25
SQL>
SQL> COLUMN first_name HEADING 'First Name' FORMAT A10
SQL> COLUMN last_name HEADING 'Last Name' FORMAT A10
SQL> COLUMN phone_no HEADING 'Phone Number' FORMAT A12
SQL>
SQL> Select first_name,last_name,phone_no
  2  from student
  3  order by last_name;

Student Phone Directory
First Name  Last Name   Phone Number
----------  ----------  ------------
Camora
Dharma
Brett
Morgan
Amanda
Thomas
Breanna
```



## X. Linking Microsoft Access to Oracle

Information on how to link a table in Oracle to a table in Access was found in an informative paper by Ivanov at George Washington University (Ivanov, 2001). It showed the steps on how to link a table from Oracle to Access. One table was linked, the STAFF table, as an example for this project.

Steps taken to link the table were as follows:

- With Microsoft Access open and in the Tables tab of the database window, select File and Get External Data
- When the link window opens, scroll down and choose ODBC Databases
- Choose the tab Machine Data Source (as this database is only on this machine) and then select the ODBC source
- Type in the Oracle user name and password
- Select the table you want to export the data from
- The tables are linked!

The other tables were not linked as the Staff table was used as an example, and the end user is not interested in linking the databases. In order to use the Staff/Board Members table at this point, the operating system Access is running on also needs to have Oracle installed. Because that is not the case at MCS, it was decided to only use the Staff/Board Members table as an example. When opening up that table or form, the user has to enter a username and password for Oracle on a machine with both Oracle and Access installed.

## XI. Comparison of Access and Oracle

Microsoft Access is an easy-to-use database for one user. It is not meant to be used by multiple users with a server like Oracle 9i is intended for. In the case of the MCS database, this works out just fine, as the organization is small and only one user uses the database at a time. Because Access is already installed on the school's computers, there is no cost to the school currently.

Comparing the development of the two databases is not easy to do, as they are created in a totally different manner. Much time and effort was put out initially to determine the correct relationships in this database. Creating the tables was quick and easy, as Access provides so many tools to help with this process. It was found that if changes were made to a table that was linked to another table, creating a relationship, the relationship had to be deleted before the changes could be made.

The Oracle database was created after the Access database, so the relationships were already determined and working. With the relational model in place and working well for Access, it was relatively easy to write the code for Oracle. Creating and then populating the tables took the most time. Populating the tables in Oracle took more time and careful typing, as it was necessarily to type the information in the correct order. Populating tables in Access was much easier with the use of columns and headings.

The most important part of working with the databases was the creation of relationships and making sure the tables were connected correctly.

In researching Oracle products, there is an Oracle 10g available for small-to-medium businesses. It is relatively inexpensive at $149 for a single user (Oracle, 2004). It appears to be easy to install and use.

## XII. Results and Conclusion

The Madison Christian School database meets the requirements and objectives that were set out for it. It provides the necessary information that MCS needs to use on a daily basis. It will be of great assistance when it comes time to order textbooks each summer and generate reports for the school year. The Access Forms and Reports are easy to access and use. Information can now be manipulated and added with little effort. Access is familiar to the end users and works well for MCS.

Developing the database started with researching the school. Organizational information was gathered, analyzed, and use to create the relational models.

The Oracle database was developed with the growth of the school in mind. If the school ever goes to a server-side environment, Oracle would be a good option to explore implementing.

There is great opportunity to expand the database as the school grows and develops. The database could expand to include the following:

- Room assignments
- Inventory
- Sponsor information
- Staff information could be expanded as staff is added

The project coordinator learned a great deal about creating a database that meets the requirements of the end user. It is extremely important to plan before starting the development of the database. The ER diagram and relational model need to be carefully thought out and developed before the development of the database, which proved to be of great assistance during creation. Planning is key when creating a database. Initial

interviews and follow-up interviews provided much needed information on what the end users expected and needed in this database.

The end users are pleased with the database and look forward to using it throughout this year and into the future.

## XIII. References

Ivanov, Sergey (2001). *Access-ODBC-Oracle 8i Connect Instructions*. Retrieved October 12, 2004 from http://msist-web.va.gwu.edu/Oracle/Instructions/Access-Oracle-ODBC-Connect.pdf

Oracle Data Sheet (January 2004). *Oracle Database 10G Standard Edition One.* Retrieved October 15, 2004 from http://www.oracle.com/database/Std_One.html

## APPENDIX A – Interview Questionnaire

The following questionnaire and answers were used during an initial interview with the administrator/head teacher.

Interview with: Lisa Sum
Date: September 7, 2004

1. Tell me about your role at MCS.

I am the head teacher here at MCS. Because we are a small school, I also do administrative tasks as needed. I work closely with the Board of Directors and am accountable to them. I teach language arts and history for grades 4-8. There are three other full time teachers and three part time teachers.

2. What are your objectives for the school year?

I want to provide a quality education for our students in a Christian environment.

3. What issues or problems do you face at MCS?

The administrative tasks are my biggest area of frustration. I like to concentrate on teaching but often get bogged down with administration. There is always paperwork to file, fill out, or send out to parents.

4. What people, places and things are of interest to you in your job?

The students are my biggest interest. I also have to be keenly aware of curriculum needs. I need to know where a given student is at a given time.

5. What information do you look back at from previous years?

I look back on previous addresses and grades made in a certain subject.

6. What information do you use on a daily basis, a weekly basis, or once a year?

Daily I use information on students' whereabouts and what subjects they are taking. I use my lesson plan book and my curriculum. Weekly I use the student/parent directory for phone numbers and addresses. Also, we have a weekly teacher's meeting that I use all kinds of information for including curriculum and student information. Yearly I will look at grades received in a subject from the previous year.

## APPENDIX B – Microsoft Access Reports

Below are examples of reports generated through the MCS Database. Only page one of each report is shown for example purposes.

# *Books used by All Grades*

| *Subject Class Level Title* | *Subject Name* | *Publisher ID* | *Book ID* | *Book* |
|---|---|---|---|---|
| **Fifth** | | | | |
| | *5 Language* | | | |
| | | Houghton Mifflin | 0-618-09064-9 | Workbooks |
| | | Houghton Mifflin | 0-618-03082-4 | Student |
| | *5 Math* | | | |
| | | ACSI | 7218 | Student |
| | *5 Reading* | | | |
| | | Bob Jones | 125617 | Student |
| | | Bob Jones | 125633 | Workbook |
| | *5 Science* | | | |
| | | Concordia Publishing House | 52-1005 | Student |
| | | Concordia Publishing House | 52-3143 | Workbook |
| | *5 Social Studies* | | | |
| | | Abeka | 61344 | Student |
| | | Abeka | 61387 | Assessment |
| | | Abeka | 61409 | Quizes |
| | | Abeka | 61360 | Workbooks |
| | *5 Spelling* | | | |
| | | ACSI | 7415 | Spelling |
| **First** | | | | |
| | *1 Bible* | | | |
| | | ACSI | 7114 | Bible |
| | *1 Handwriting* | | | |
| | | Bob Jones | 114363 | Handwriting |
| | *1 Language* | | | |
| | | Houghton Mifflin | 1-45011 | Lang book |
| | *1 Math* | | | |

# *Student*

| *Grade* | *Last Name* | *First Name* | *Street* | *City* | *State* | *Zip* | *Phone Number* | *Birthdate* |
|---|---|---|---|---|---|---|---|---|
| *1* | | | | | | | | |
| | | Dharma | | Madison | SD | 57042- | | |
| | | Brett | | Madison | SD | 57042- | | |
| | | April | | Madison | SD | 57042- | | |
| | | Gabriel | | Madison | SD | 57042- | | |
| | | Taylor | | Madison | SD | 57042- | | |
| *2* | | | | | | | | |
| | | Morgan | | Madison | SD | 57042- | | |
| | | Hanna | | Madison | SD | 57042- | | |
| | | Paige | | Madison | SD | 57042- | | |
| | | Jonny | | Madison | SD | 57042- | | |
| | | Grant | | Madison | SD | 57042- | | |
| | | Drake | | Madison | SD | 57042- | | |
| | | Bethany | | Madison | SD | 57042- | | |
| *3* | | | | | | | | |
| | | Jackson | | Madison | SD | 57042- | | |
| | | Thomas | | Madison | SD | 57042- | | |
| | | Abrienne | | Madison | SD | 57042- | | |
| *4* | | | | | | | | |
| | | Daniel | | Madison | SD | 57042- | | |

# *Classes Taught by Teacher*

| *Last Name* | *Subject Class Level* | *Subject Name* |
|---|---|---|
| **Bekker** | | |
| | *Kindergarten* | |
| | | K math |
| | | K reading |
| | | K science |
| | | K Social Studies |
| | *Third* | |
| | | 3 Bible |
| | | 3 Handwriting |
| | | 3 Language |
| | | 3 Math |
| | | 3 Reading |
| | | 3 Science |
| | | 3 Social Studies |
| | | 3 Spelling |
| **Goodal** | | |
| | *Fifth* | |
| | | 5 Bible |
| | | 5 Math |
| | | 5 Science |
| | *Fourth* | |
| | | 4 Bible |
| | | 4 Math |
| | | 4 Science |
| | *Sixth* | |
| | | 6 Bible |
| | | 6 Math |
| | | 6 Science |
| **Roswell** | | |
| | *First* | |
| | | 1 Bible |

## APPENDIX C – Oracle SQL code

Code for creating tables

```
create table faculty
(faculty_id number(6),
first_name varchar2(20),
mid_name varchar2(20),
last_name varchar2(25),
street varchar2(30),
city varchar2(25),
state char(2),
zip char(5),
phone_no char(10),
gender char(1),
birthdate date,
hire_date date,
certification varchar2(30),
constraint faculty_faculty_id_pk primary key (faculty_id),
constraint faculty_gender_cc check ((gender = 'm' ) or (gender = 'f')));

create table student
(student_id number(6),
first_name varchar2(20),
mid_name varchar2(20),
last_name varchar2(25),
street varchar2(30),
city varchar2(25),
state char(2),
zip char(5),
phone_no char(10),
gender char(1),
birthdate date,
matriculation_date date,
matriculated_class_level varchar2(10),
current_class_level varchar2(10),
guardian_first_name varchar2(20),
guardian_last_name varchar2(20),
guardian_relationship varchar2(20),
guardian_email varchar2(20),
constraint student_student_id_pk primary key (student_id),
constraint student_gender_cc check ((gender = 'm' ) or (gender = 'f')));

create table staff
(staff_id number(6),
first_name varchar2(20),
mid_name varchar2(20),
```

```
last_name varchar2(25),
street varchar2(30),
city varchar2(25),
state char(2),
zip char(5),
phone_no char(10),
gender char(1),
birthdate date,
appointed_date date,
position varchar2(30),
constraint staff_staff_id_pk primary key (staff_id),
constraint staff_gender_cc check ((gender = 'm' ) or (gender = 'f')));


create table subject
(subject_id number(5),
subject_class_level varchar2(2),
subject_name varchar2(25),
constraint subject_subject_id_pk primary key (subject_id));


create table teaching
(faculty_id number(6),
subject_id number(5),
constraint t_faculty_id_Subj_id_pk primary key (faculty_id,
subject_id),
constraint teaching_faculty_id_fk foreign key (faculty_id)
references faculty (faculty_id),
constraint teaching_subject_id_fk foreign key (subject_id)
references subject (subject_id));

create table publisher
(publisher_id number(6),
publisher_name varchar2(20),
mcs_account_number varchar2(20),
phone_no char(20),
web_address varchar2(20),
constraint publisher_publisher_id_pk primary key (publisher_id));

create table book
(book_id varchar2(20),
subject_id number(5),
book_title varchar2(30),
publisher_id number(6),
constraint book_book_id_pk primary key (book_id),
constraint book_subject_id_fk foreign key (subject_id)
```

```
references subject (subject_id),
constraint book_publisher_id_fk foreign key (publisher_id)
references publisher (publisher_id));

create table studies
(student_id number(6),
subject_id number(5),
constraint s_student_id_Subj_id_pk primary key (student_id,
subject_id),
constraint studies_student_id_fk foreign key (student_id)
references student (student_id),
constraint studies_subject_id_fk foreign key (subject_id)
references subject (subject_id));
```

Code to populate tables (Redacted for printing)

```
insert into faculty values
(200, 'Keri', null, 'Bekker', xxxx', 'Madison', 'SD', '57042','xxxxx','f',to_date('06-04-
1975','MM-DD-YYYY'),to_date('07-01-2004','MM-DD-YYYY'),'Music/MS Language');

insert into faculty values
(201, 'Jill', null, 'Roswell', 'xxxxx', 'Brookings', 'SD', '57006','xxxxx','f',to_date('11-11-
1960','MM-DD-YYYY'),to_date('07-01-2003','MM-DD-YYYY'),'Elementary');

insert into faculty values
(202, 'Dan', null, 'Goodal', 'xxxxx', 'Madison', 'SD', '57042','xxxxx','m',to_date('02-17-
1960','MM-DD-YYYY'),to_date('07-01-2003','MM-DD-YYYY'),'Secondary');

insert into faculty values
(203, 'Lisa', null, 'Sum', 'xxxxx', 'Madison', 'SD', '57042','xxxxx','f',to_date('12-31-
1960','MM-DD-YYYY'),to_date('07-01-2000','MM-DD-YYYY'),'Secondary');

insert into faculty values
(204, 'Tim', null, 'Trent', 'xxxxx', 'Madison', 'SD', '57042','xxxxx','m',to_date('06-30-
1975','MM-DD-YYYY'),to_date('08-01-2004','MM-DD-YYYY'),'Secondary');

insert into subject values
(1,'K','KReading');

insert into subject values
(2,'K','KMath');

insert into subject values
(3,'K','KScience');

insert into subject values
```

```
(4,'1','1Reading');

insert into subject values
(5,'1','1Math');

insert into subject values
(6,'2','2Science');

insert into subject values
(7,'3','3Reading');

insert into subject values
(8,'3','3Math');

insert into subject values
(9,'3','3Science');

insert into subject values
(10,'4','4Reading');

insert into subject values
(11,'4','4Math');

insert into subject values
(12,'4','4Science');

insert into subject values
(13,'5','5Reading');

insert into subject values
(14,'5','5Math');

insert into subject values
(15,'5','5Science');

insert into subject values
(16,'6','6Reading');

insert into subject values
(17,'6','6Math');

insert into subject values
(18,'6','6Science');

insert into publisher values
(1,'Modern Curriculum','55438','18003213106',null);
```

```
insert into publisher values
(2, 'ABeka','20560354','18773569385','www.abeka.com');

insert into publisher values
(3, 'ACSI', '72898','18003670798','www.acsi.org');

insert into publisher values
(4,'Bob Jones','88908','18008455731','www.bobjones.com');

insert into publisher values
(5,'Houghton Mifflin','1638','18007332828',null);

insert into publisher values
(6, 'Concordia','9456','18004909889',null);

insert into book values
(1-56704, 1, 'All Aboard', 1);

insert into book values
(54348, 3, 'K Science',2);

insert into book values
(156704,4,'First Reading',1);

insert into book values
(7210,5,'First Math',3);

insert into book values
(52-1002,6,'Second Science',6);

insert into book values
(120766, 7, 'Third Reading',4);

insert into book values
(7212,8,'Third Math',3);

insert into book values
(52-3141,9,'Third Science',6);

insert into book values
(122796,10,'Fourth Reading',4);

insert into book values
(7214,11,'Fourth Math',3);
```

```
insert into book values
(52-1004,12,'Fourth Science',6);

insert into book values
(125617,13,'Fifth Reading',4);

insert into book values
(7218,14,'Fifth Math',3);

insert into book values
(52-3143,15,'Fifth Science',6);

insert into book values
(125666,16,'Sixth Reading',4);

insert into book values
(7219,17,'Sixth Math',3)

insert into book values
(52-3144, 18,'Sixth Science',6);
```

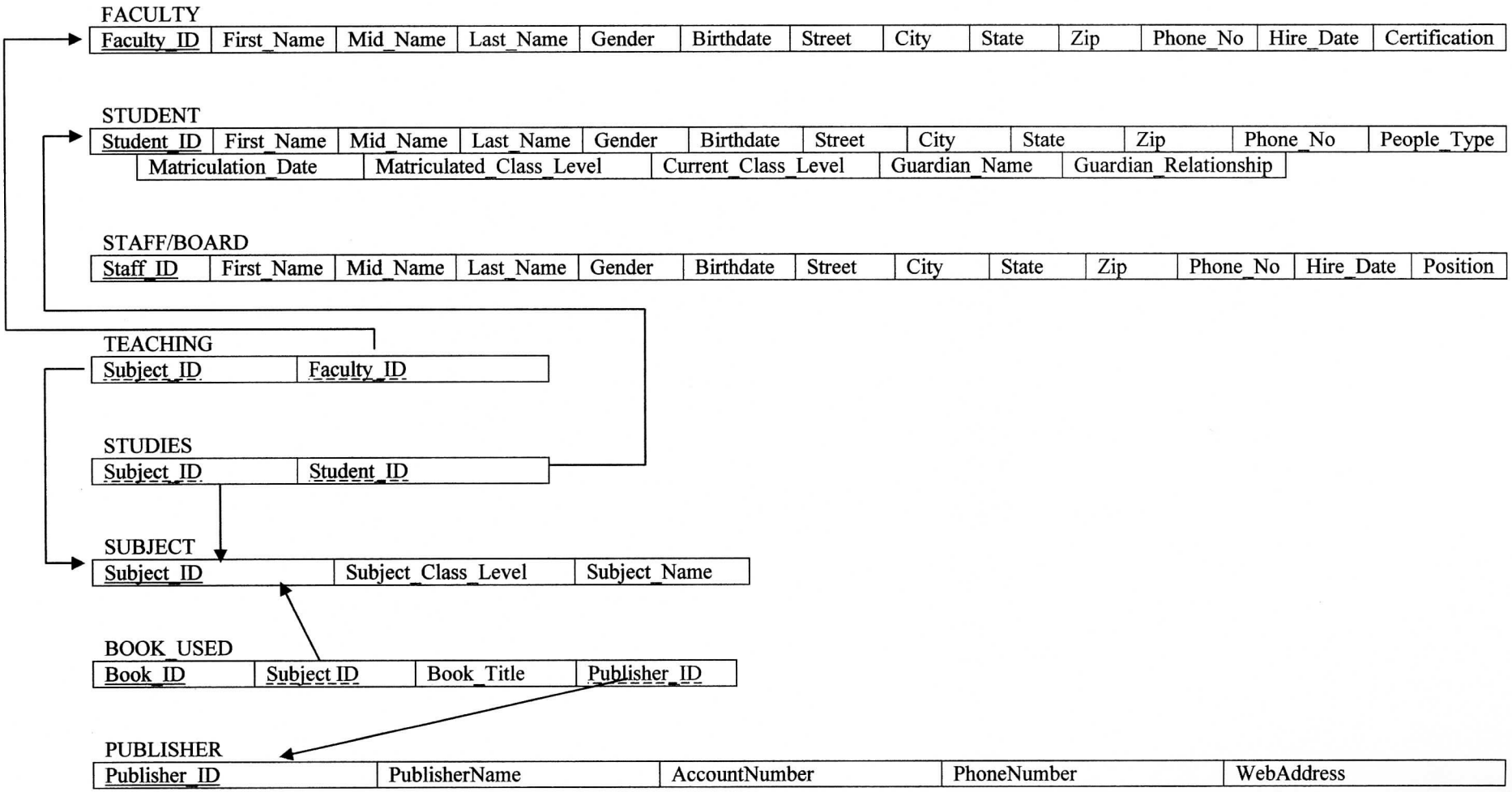
Code to run queries and create a report

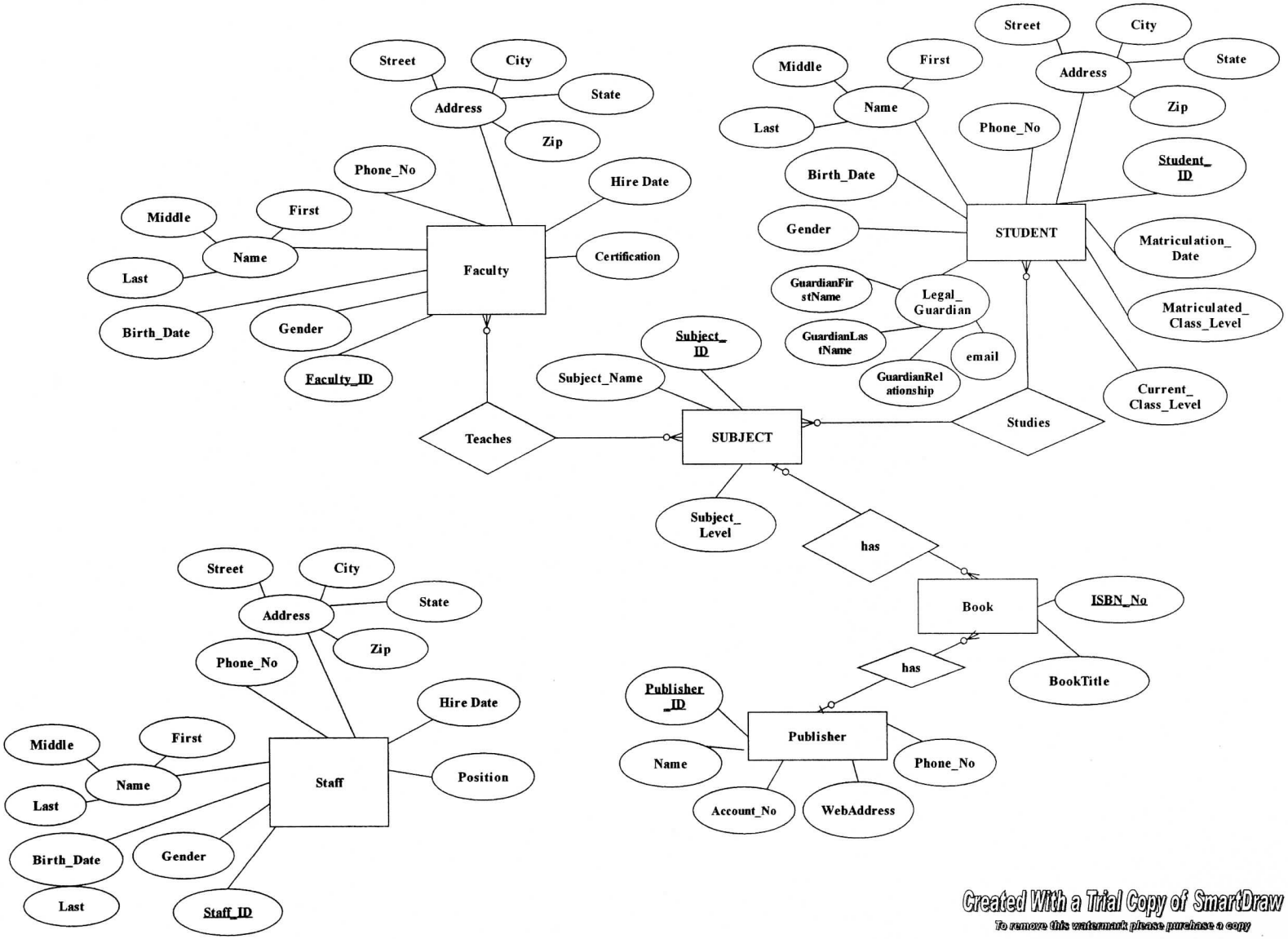
```
--Query 1 Select all students first and last names in the first grade

SELECT student_id, first_name, last_name
FROM student
WHERE current_class_level='first';


--Query 2 select subject name, book used, and publisher for grade 1 math
SELECT subject_name,book_title,publisher_name
FROM subject,book,publisher
WHERE subject.subject_id=book.subject_id
AND book.publisher_id=publisher.publisher_id
AND subject_name='1Math';


--Query 3 select student name, guardian's name, and phone number
SELECT first_name, last_name, guardian_first_name, guardian_last_name, phone_no
FROM student
order by last_name;


--Creating a report
TTITLE LEFT "Student Phone Directory" -
```

```
Left 'Student'
SET LINESIZE 100
SET PAGESIZE 25

COLUMN first_name HEADING 'First Name' FORMAT A10
COLUMN last_name HEADING 'Last Name' FORMAT A10
COLUMN phone_no HEADING 'Phone Number' FORMAT A12

Select first_name,last_name,phone_no
from student
order by last_name;
```

## APPENDIX D – ER Diagram and Data Dictionary

Relational Model and ER Diagram





**Entities Types**

1. Name: FACULTY
   Type: Strong Entity
   Definition: A FACULTY teaches a SUBJECT, which involves explaining a specific set of issues or skills to the STUDENTS who attend that class. Each FACULTY member teaches a minimum of one SUBJECT, and often teaches more than one.
   Identifier: Faculty_ID

2. Name: STAFF
   Type: Strong Entity
   Definition: A STAFF/BOARD member is employed by or volunteers at MCS and is not a FACULTY member.
   Identifier: Staff_ID

3. Name: STUDENT
   Type: Strong Entity
   Definition: A STUDENT is an individual attending MCS for the purpose of receiving an education. STUDENTs are generally from 5 to 14 years of age. All STUDENTs study SUBJECTS that are taught by the FACULTY.
   Identifier: Student_ID

4. Name: SUBJECT
   Type: Strong Entity
   Definition: A SUBJECT is a set of skills or an area of knowledge that is taught by FACULTY to STUDENTs during classes. Each SUBJECT at MCS is subdivided not only by Subject_ID, but also by Subject_Class_Level, and Subject_Name.
   Identifier: Subject_ID

5. Name: BOOK
   Type: Strong Entity
   Definition: A BOOK is used by each STUDENT in a class. A FACUTLY teaches from the BOOK. Each BOOK has a PUBLISHER and a Book_ID number.
   Identifier: Book_ID

6. Name: PUBLISHER
   Type: Strong Entity
   Definition: A BOOK is published by a PUBLISHER. The PUBLISHER has a phone number and web address. MCS has an account number with each PUBLISHER.
   Identifier: Publisher_ID

**Relationship Types**

1. Name: Teaches
   Type: Binary M:N
   Description: A FACULTY teaches a SUBJECT, which involves explaining a specific set of issues or skills to the STUDENTS who attend that class. Each FACULTY member teaches a minimum of zero SUBJECTs, and often teaches more than one.

**Dictionary of Entity Attributes**

**Faculty**

1. Name: Faculty_ID
Definition: This unique, non-repeating, alphanumeric value is assigned to all faculty associated to MCS. The Faculty_ID serves as the unique identifier for a Faculty.
Null: no
2. Name: Name
Definition: A person's full name, this composite attribute has three other attributes associated with it: First, Middle, and Last.
Null: no (Middle Name is allowed to be null)
3. Name: Birth_Date
Definition: The date on which a person was born.
Null: no
4. Name: Gender
Definition: The gender of a person (male or female).
Null: no
5. Name: Address
Definition: A person's full address, this composite attribute has four other attributes associated with it: Street, City, State, and Zip.
Null: no
6. Name: Phone_No
Definition: A person's telephone number.
Null: no
7. Name: Hire_Date
Definition: The date a faculty member was hired
Null: no
8. Name: Certification
Definition: Indicates the certification a faculty member has
Null: no

**Staff/Board Member**

1. Name: Staff_ID
Definition: This value serves as the unique identifier of any Staff or Board Member
Null: no
2. Name: Name
Definition: A person's full name, this composite attribute has three other attributes associated with it: First, Middle, and Last.
Null: no (Middle Name is allowed to be null)
3. Name: Birth_Date
Definition: The date on which a person was born.
Null: no
4. Name: Gender
Definition: The gender of a person (male or female).
Null: no
5. Name: Address
Definition: A person's full address, this composite attribute has four other attributes associated with it: Street, City, State, and Zip.
Null: no
6. Name: Phone_No
Definition: A person's telephone number.
Null: no
7. Name: Hire_Date
Definition: The date a faculty member was hired
Null: no
8. Name: Position
Definition: Indicates the position a staff or board member has
Null: no

**STUDENT**

1. Name: Student_ID
Definition: This value serves as the unique identifier of any STUDENT, past or present. This number is typically used for the purposes of scheduling classes and keeping records.
Null: no
2. Name: Name
Definition: A person's full name, this composite attribute has three other attributes associated with it: First, Middle, and Last.
Null: no (Middle Name is allowed to be null)
3. Name: Birth_Date
Definition: The date on which a person was born.
Null: no
4. Name: Gender
Definition: The gender of a person (male or female).

Null: no
5. Name: Address
Definition: A person's full address, this composite attribute has four other attributes associated with it: Street, City, State, and Zip.
Null: no
6. Name: Phone_No
Definition: A person's telephone number.
Null: no
7. Name: Matriculation_Date
Definition: This date value shows when a student joined the school.
Null: no
8. Name: Matriculated_Class_Level
Definition: This attribute shows what a student's Current_Class_Level.
Null: no
9. Name: Current_Class_Level
Definition: This derived attribute shows what "grade" a student is currently in.
Null: yes (can enter a students information without an assigned grade level)
10. Name: Guardian_Name
Definition: This attribute of student holds the name of that student's parent or official family contact person for the purpose of legal correspondence, grade reporting, illness, or other appropriate situations. The database will hold one value for this attribute.
Null: No
11. Name: Guardian_Relationship
Definition: This attribute of student will tell what relationship the student has to the legal guardian.
Null: No

**SUBJECT**

1. Name: Subject_ID
Definition: Each subject will vary by relevant material and also by level of difficulty; this number will distinguish all such instances of SUBJECT.
Null: No
2. Name: Subject_Name
Definition: This attribute lists the common name of a specific course's subject matter. Examples include math, science, or band.
Null: No
3. Name: Subject_Class_Level
Definition: This attribute lists the Class_Level associated with the degree of difficulty of the subject material.
Null: No

**BOOK**

1. Name: Book_ID
Definition: This attribute is the unique identifier for a book used for a subject.
Null: no
2. Name: Book_Title
Definition: The title of the book.
Null: no
3. Name: Subject_ID
Definition: Linked to Subject table
Null: no
4. Name: Publisher_ID
Definition: Linked to Publisher table
Null: no

**PUBLISHER**

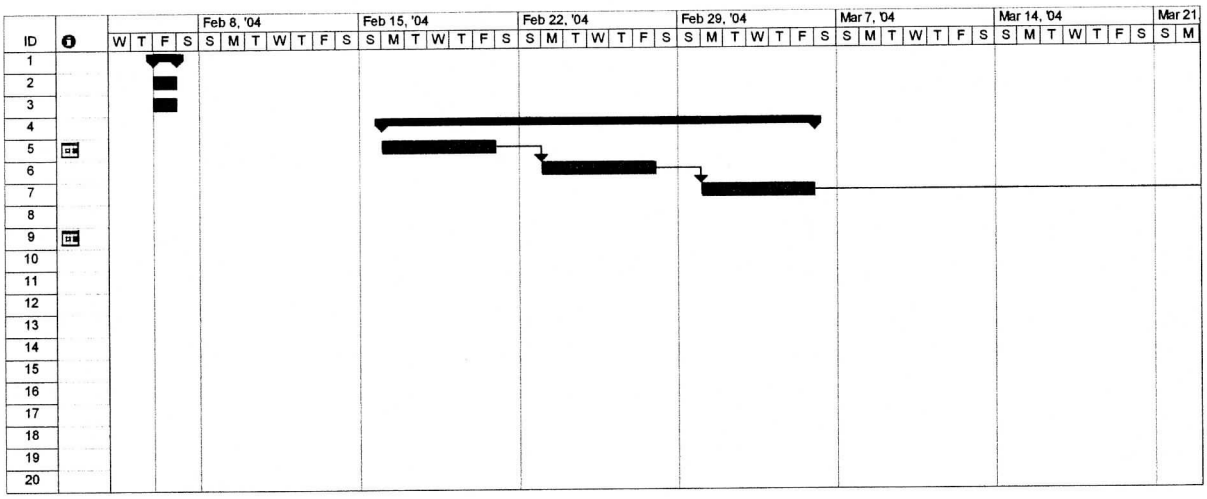
1. Name: Publisher_ID
Definition: This attribute is the unique identifier for a publisher of a book.
Null: no
2. Name: Publisher_Name
Definition: The name of the publisher.
Null: no
3. Name: Account_Number
Definition: The account number assigned by the publisher to MCS.
Null: no
4. Name: PhoneNumber
Definition: The phone number of the publisher
Null: yes
5. Name: WebAddress
Definition: The web address of the publisher
Null: yes

## APPENDIX E – WBS and Gantt Chart

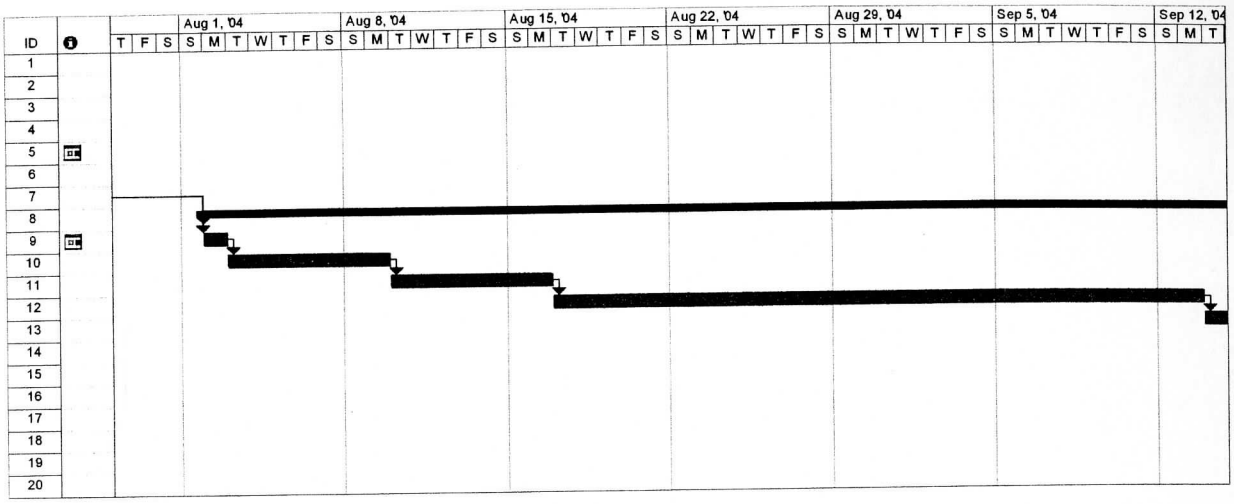
Work Breakdown Structure and Gantt Chart

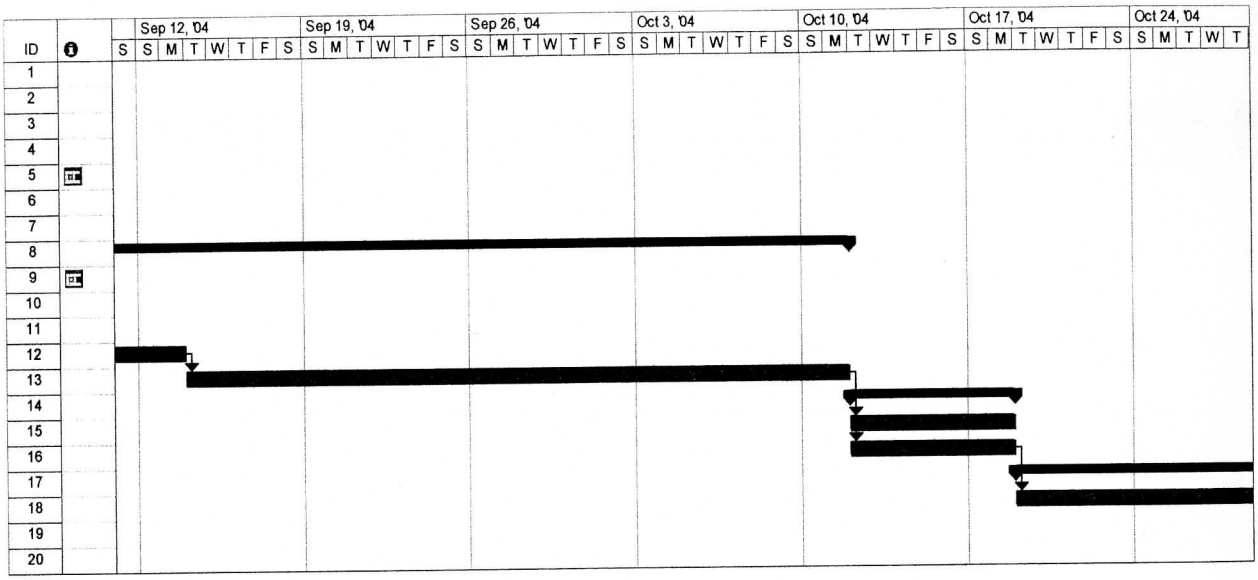
Below I have outlined a work breakdown structure and Gantt Chart for development of the database systems. As you will note, I have opted to take the summer off from development due to family commitments. This project will be completed in time for a December 2004 graduation date.

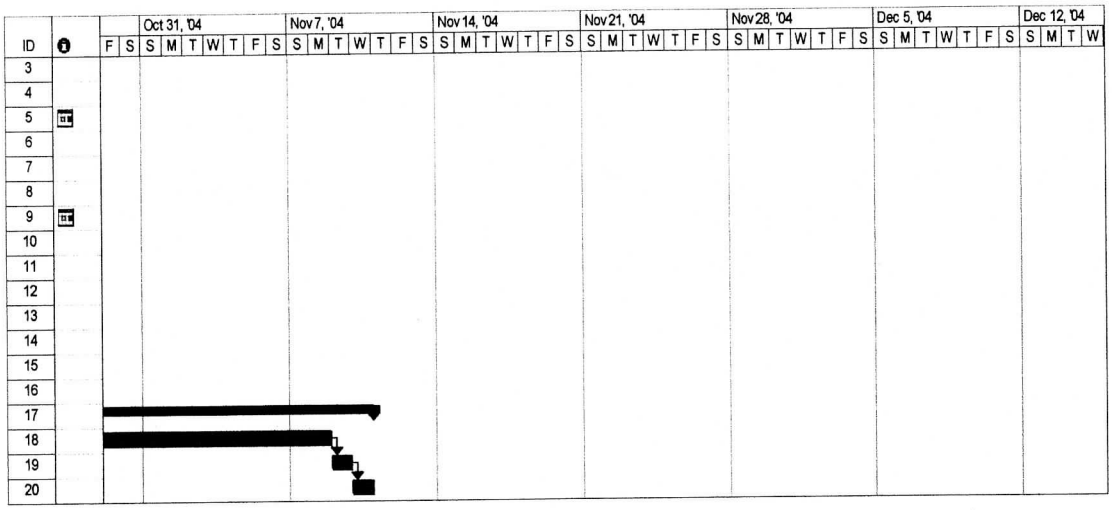
| ID | Task Name | Duration | Start | Finish | Predecessors | Resource Names |
|---|---|---|---|---|---|---|
| 1 | **1 Initializing** | **1 day** | **Fri 2/6/04** | **Fri 2/6/04** | | |
| 2 | 1.1 Project Idea Form Devleopm | 1 day | Fri 2/6/04 | Fri 2/6/04 | | |
| 3 | 1.2 Submission of Project Idea | 1 day | Fri 2/6/04 | Fri 2/6/04 | | |
| 4 | **2 Planning** | **15 days** | **Mon 2/16/04** | **Fri 3/5/04** | | |
| 5 | 2.1 Develop project plans | 1 wk | Mon 2/16/04 | Fri 2/20/04 | | |
| 6 | 2.2 Review project plans | 1 wk | Mon 2/23/04 | Fri 2/27/04 | 5 | |
| 7 | 2.3 Project plans approved | 1 wk | Mon 3/1/04 | Fri 3/5/04 | 6 | |
| 8 | **3 Executing** | **51 days** | **Mon 8/2/04** | **Mon 10/11/04** | | |
| 9 | 3.1 Installation of Oracle and Ac | 1 day | Mon 8/2/04 | Mon 8/2/04 | 7 | |
| 10 | 3.2 Cost Analysis of each | 1 wk | Tue 8/3/04 | Mon 8/9/04 | 9 | |
| 11 | 3.3 Developing Database Desigı | 1 wk | Tue 8/10/04 | Mon 8/16/04 | 10 | |
| 12 | 3.4 Implementation of Design in | 4 wks | Tue 8/17/04 | Mon 9/13/04 | 11 | |
| 13 | 3.5 Implemenation of Design in | 4 wks | Tue 9/14/04 | Mon 10/11/04 | 12 | |
| 14 | **4 Controlling and Testing of Databas** | **5 days** | **Tue 10/12/04** | **Mon 10/18/04** | | |
| 15 | 4.1 Report performance | 1 wk | Tue 10/12/04 | Mon 10/18/04 | 13 | |
| 16 | 4.2 Contrast and Comparision of | 1 wk | Tue 10/12/04 | Mon 10/18/04 | 13 | |
| 17 | **5 Closing** | **17 days** | **Tue 10/19/04** | **Wed 11/10/04** | | |
| 18 | 5.1 Prepare final project report | 3 wks | Tue 10/19/04 | Mon 11/8/04 | 16 | |
| 19 | 5.2 Present final project | 1 day | Tue 11/9/04 | Tue 11/9/04 | 18 | |
| 20 | 5.3 Project completed | 1 day | Wed 11/10/04 | Wed 11/10/04 | 19 | |




SUMMER BREAK!


ID
Aug 1, '04
Aug 8, '04
Aug 15, '04
Aug 22, '04
Aug 29, '04
Sep 5, '04
Sep 12, '04


ID
Sep 12, '04
Sep 19, '04
Sep 26, '04
Oct 3, '04
Oct 10, '04
Oct 17, '04
Oct 24, '04


ID
Oct 31, '04
Nov 7, '04
Nov 14, '04
Nov 21, '04
Nov 28, '04
Dec 5, '04
Dec 12, '04